倍精度演算を用いた 4 倍精度演算環境の実装と効果

Implementation and effect of quadruple precision arithmetic using double precision

[1,*] 齊藤 翼 [2] 石渡 恵美子 [3] 長谷川 秀彦
1) 東京理科大学大学院，2) 東京理科大学，3) 筑波大学

[1]Tsubasa Saito [2]Emiko Ishiwata [3]Hidehiko Hasegawa
[1]Graduate School of Science, Tokyo University of Science,
[2]Tokyo University of Science, [3]University of Tsukuba
*Email: j1409612@ed.kagu.tus.ac.jp

キーワード：4 倍精度演算，丸め誤差，一般化共役残差法，リスタート，Scilab
Keywords: quadruple precision arithmetic, round-off error, GCR method, restart, Scilab

1 はじめに

コンピュータにおける数値計算で用いられる浮動小数演算では，桁落ちや丸め誤差の影響は避けられない．これらを克服するためには，浮動小数点数のビット数を増やすか，特別なアルゴリズムが必要となる．

本研究では，Dekker [1] らによる丸め誤差を考慮した浮動小数点数アルゴリズムに基づき，倍精度演算の組み合わせで計算される 4 倍精度 Double-double 演算 (DD と記す [2]) を Scilab [3] に実装し，悪条件な連立一次方程式に一般化共役残差（Generalized Conjugate Residual, GCR）法を適用し，多倍長演算の有効性を示す.

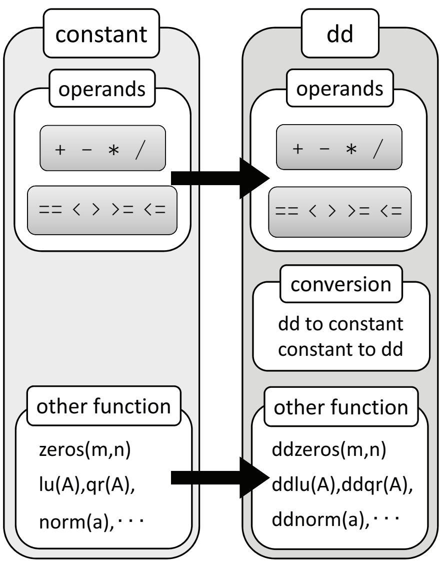

図 1: Scilab 上での DD への拡張

2 DD の実装

DD では，数 α を 2 つの倍精度浮動小数点数 $A.hi$ と $A.lo$ を用いて，以下のように表す：

$A.hi$ = (αを倍精度に丸めた値)
$A.lo$ = (($\alpha - A.hi$) を倍精度に丸めた値)

われわれの実装では，Scilab に 2 つの constant を持つ新しい型 ”dd” を設定し，四則演算の多重定義や，既存の関数の拡張を行った．この結果，DD を従来の Scilab の記述と矛盾なく導入できた（図 1）.

3 DD を用いた GCR(m) 法の特性解析

連立一次方程式 $A\boldsymbol{x} = \boldsymbol{b}$ の反復解法である GCR(m) 法（m はリスタート周期）は，各反復で残差ノルムが減少し，高々次元数の反復で解に収束する．しかし，浮動小数演算では残差が停滞する場合や，残差は減少しても誤差が減少していない場合などが知られている.

今回，悪条件の行列 A として，非零成分が $a_{i+2,i} = \gamma$，$a_{i,i} = 2$，$a_{i,i+1} = 1$ の Toeplitz 行列を選んだ．右辺ベクトルは，解 $\boldsymbol{x}^*$ の値がすべて 1 となるように定め，初期ベクトル $\boldsymbol{x}_0 = \boldsymbol{0}$ とし，$N = 50$ とした.

$m = \infty$（リスタートなし）とし，反復 50 回目の相対残差ノルムを図 2 に示す．これから，本来の倍精度の演算精度で計算でき

るのは，停止条件を $\|\boldsymbol{r}_k\|_2 \le 10^{-12}\|\boldsymbol{r}_0\|_2$ としたとき $\gamma = 3.4$，条件数 8.1×10^3 までであり，DD を用いれば $\gamma = 5.8$，条件数 2.6×10^8 まで収束することがわかる．

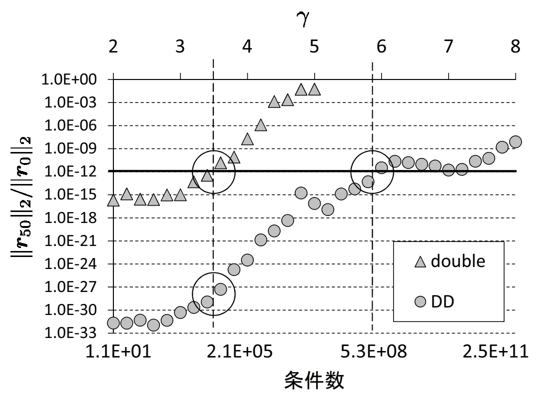

図 2: 50 反復における条件数と相対残差

次にリスタート周期 m を N に近い値とした．$\gamma = 7$ のときの DD の収束履歴を図 3 に示す．停止条件を $\|\boldsymbol{r}_k\|_2 \le 10^{-12}\|\boldsymbol{r}_0\|_2$ としたとき，$m = \infty$ では収束しないのに対して，$m = 47$ は 231 回，$m = 50$ は 89 回，$m = 53$ は 96 回の反復で収束した．$m = \infty, 50, 53$ では 50 反復付近で残差は急落し，リスタート後の 50 反復で再び残差が減少した．$m = 47$ の場合は残差の減少が小さい．

停止条件を $\|\boldsymbol{r}_k\|_2 \le 10^{-24}\|\boldsymbol{r}_0\|_2$ としたとき，$m = 47$ は 328 回，$m = 50$ は 148 回，$m = 53$ は 154 回の反復で収束した．$m = \infty, 50, 53$ では 50 反復付近での相対残差ノルムは 1.42×10^{-12} 程度となる．しかし $m = 50, 53$ は，リスタート後の 50 反復で再び残差が大きく減少する．

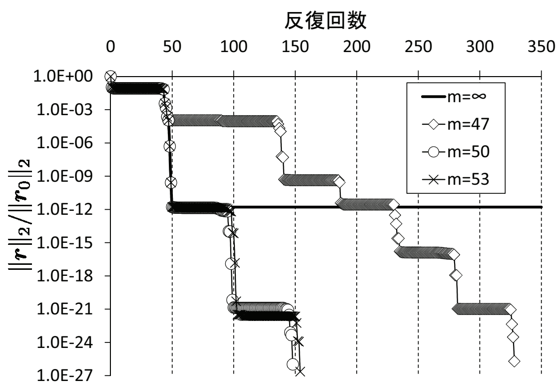

図 3: 収束のリスタート周期依存性

このように，$m = \infty$ では収束しない問題に対して，リスタート周期を $m \ge N$ とすると，残差の減少に大きな効果がある．

上記の結果をふまえて $m = 50$，停止条件を $\|\boldsymbol{r}_k\|_2 \le 10^{-12}\|\boldsymbol{r}_0\|_2$ とした．倍精度では，$\gamma = 4$（条件数 2.1×10^5）では収束せず，相対誤差ノルムは 3.05×10^{-4} となる．DD の場合は $\gamma = 4$ あたりでは精度の良い解を得られるが，$\gamma = 8$（条件数 2.5×10^{11}）のとき 98 反復で収束するものの，相対誤差は 5.61×10^{-3} となる．

停止条件をそのままで，DD を用いることで，倍精度では収束しにくい問題に対して収束性が向上するが，倍精度の場合と同様に，条件数が大きくなると誤差が減少しない．これに対して $\gamma = 8$ のとき停止条件を $\|\boldsymbol{r}_k\|_2 \le 10^{-24}\|\boldsymbol{r}_0\|_2$ のように厳しくすると，196 回の反復で相対誤差は 2.84×10^{-15} まで改善される．

4 まとめ

反復解法は浮動小数演算では悪条件な問題に対して残差が停滞してしまう傾向があった．これに対し DD を用いて，GCR(m) 法のリスタート周期を $m \ge N$ としたとき，残差がさらに減少することがわかった．DD を用いても，条件数が大きくなると誤差が減少しない傾向は，倍精度と変わらないこともわかり，環境に応じて停止条件を設定する必要があるといえる．

われわれは DD を Scilab に実装した．演算の多重定義や，既存の関数の拡張により，従来の Scilab のコードとほぼ同様の記述で DD を利用することができる．これを使うことで，浮動小数演算による丸め誤差の影響を強く受けるアルゴリズムの検証を手軽に行うことができる．

参考文献

[1] T.J. Dekker, A Floating-Point Technique for Extending the Available Precision, Numer. Math. Vol. 18 (1971) 224-242.

[2] D.H.Bailey,DDFUN90, http://crd.lbl.gov/~dhbailey/mpdist/

[3] Scilab, http://www.scilab.org/